

# Distributed Estimation for Cooperative Mobile Manipulation

Antonio Franchi, Antonio Petitti, Alessandro Rizzo

**Abstract**—We present a distributed method for the estimation of the kinematic parameters, the dynamic parameters, and the kinematic state of an unknown body manipulated by a decentralized team of mobile ground (planar) robots. The proposed approach relies on the geometry of the rigid body kinematics, the rigid body dynamics, on nonlinear observation, and on consensus algorithms. The only three requirements are that each robot is able to control the 2D wrench exerted locally on the load, it can measure the velocity of its contact point, and the communication graph is connected. The finite time convergence of the strategy is proven and all the robots agree on the same estimated quantities at the end of the procedure. We present also two basic distributed control strategies that are proven to satisfy nonlinear observability conditions needed for the estimation accomplishment. Finally, a numerical test that demonstrates the evolution of the estimation algorithm is given.

## I. INTRODUCTION

In this paper, we propose what we believe is the first fully-distributed method for the estimation of all the quantities and parameters needed by a team of ground (planar) mobile robots to collectively manipulate an unknown load. In particular, the proposed algorithm provides the estimation of the kinematic parameters (equivalent to the grasp matrix), the dynamic parameters (relative position of the center of mass, mass, and rotational inertia) and the kinematic state of the load (velocity of the center of mass and rotational rate).

Most of the work presented in the literature is based on the assumption of the a-priori knowledge of the inertial parameters of the load, although this assumption does not always hold in real-world scenarios [1]–[5]. Thus, collective manipulation tasks would benefit from the implementation of on-line estimation strategies of the inertial parameters of unknown loads for at least two reasons: first, existing control strategies, such as force control and pose estimation could be effectively applied with satisfactory performance and a reduced control effort. Second, time-varying loads could be effectively manipulated, toward the implementation of adaptive or event-driven control strategies in uncertain environments. Furthermore, similarly to other applications in multi-robot systems, a distributed and decentralized implementation of such estimation strategies would provide robustness and scalability.

The research on estimation of inertial parameters is at its early stage, and main limitations of the existing approaches are centralization and the use of absolute position and acceleration

measurements, which are hard and costly to achieve, especially if accurate and noise-free information is needed. Moreover, centralized strategies are notoriously poorly scalable and not robust, due to the existence of a single point of failure [6]–[9].

The algorithm that we propose has the following properties: (i) there is no central processing unit; (ii) each robot is only able to exchange information with its neighbors in the communication network; (iii) the communication network is only required to be connected (e.g., a simple line in the worst case); (iv) each robot is able only to perform local sensing and computation; (v) the amount of memory and number of computations per step needed by each local instance of the algorithm do not depend on the number of robots but only on the number of communication neighbors.

The only assumptions that are needed are that each robot is endowed with a planar manipulator that is able to exert and measure the local force applied to the load and to measure the velocity of the contact point. Any other measurement (such as, e.g., position, distance, acceleration, and gyro measurements) is not available to the robots. Furthermore, nothing is known about the manipulated load.

The approach is totally distributed, and relies on the geometry of the rigid body kinematics, the rigid body dynamics, on nonlinear observation, and on consensus strategies. It is based on a sequence of steps that is proven to converge in finite time, after which all the robots will agree on the estimation of all the following quantities, characteristic of the load: its mass, its rotational inertia, the relative position of the contact point of each robot with respect to the geometric center of the contact points, the relative position of the load center of mass with respect to the geometric center, the velocity of the center of mass and the object angular rate.

This paper builds on and expands preliminary results presented in the conference papers [10], [11]. Major improvements concern: (i) largely streamlined problem and algorithm formalizations, including a much clearer formalism for the algorithm and an explicit use of the grasp matrix, (ii) improvement of the algorithm performance (e.g., improving the estimate of the load angular velocity), (iii) consideration of the manipulation torques, (iv) local control strategies that guarantee the feasibility of the approach, and (v) a full illustrative simulation of the whole approach.

## II. MODEL AND PROBLEM STATEMENT

In this section, we formally define the problem of distributively estimating *all* the parameters and the time-varying quantities needed for a decentralized team of  $n$  ground mobile manipulators to cooperatively move an unknown *load B* mounted on a cart, as depicted in Fig. 1 from the top.

Since the load moves on the floor it is more convenient to cast the problem in 2D. Any spatially-related quantity which will be introduced in the following should be considered as the

A. Franchi is with CNRS, LAAS, 7 avenue du colonel Roche, F-31400 Toulouse, France and Univ de Toulouse, LAAS, F-31400 Toulouse, France [afranchi@laas.fr](mailto:afranchi@laas.fr)

A. Petitti is with the Institute of Intelligent Systems for Automation, National Research Council (ISSIA-CNR), 70126 Bari, Italy, [petitti@ba.issia.cnr.it](mailto:petitti@ba.issia.cnr.it)

A. Rizzo is with the Department of Mechanical and Aerospace Engineering, NYU Tandon School of Engineering, Brooklyn NY, 11201, USA, [alessandro.rizzo@nyu.edu](mailto:alessandro.rizzo@nyu.edu), and the Dipartimento di Automatica e Informatica, Politecnico di Torino, Corso Duca degli Abruzzi 24, 10129 Torino, Italy [alessandro.rizzo@polito.it](mailto:alessandro.rizzo@polito.it)

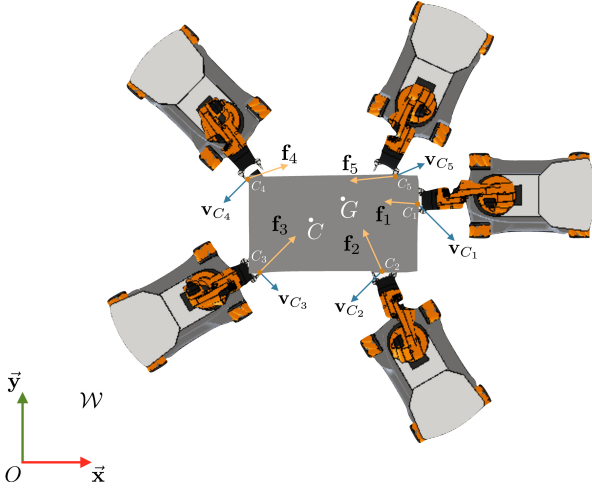


Fig. 1: Top view of a team of five mobile manipulators performing a planar manipulation task. Each robot can exert a force and torque on the object by means of a planar manipulator (only force is displayed in the picture), and can only measure the velocity of its contact point.

projection of the corresponding 3D quantity on the horizontal plane. We denote the inertial frame with  $\mathcal{W} = O - \mathbf{x}\mathbf{y}$  and the load body frame with  $\mathcal{B} = C - \mathbf{x}_B\mathbf{y}_B$ , where  $C$  is the center of mass (CoM) of  $B$ . We indicate with  $\mathbf{p}_C \in \mathbb{R}^2$  and  $\mathbf{v}_C = \dot{\mathbf{p}}_C$  the position and velocity of  $C$  expressed in  $\mathcal{W}$ , respectively, and with  $\omega \in \mathbb{R}$  the intensity of the load angular velocity, hereafter called simply *angular rate*. The dynamics of the manipulated load is the one of a rigid body

$$\mathbf{M}\dot{\mathbf{v}} + \mathbf{g} = \mathbf{u}, \quad (1)$$

where  $\mathbf{v} = (\mathbf{v}_C^T \omega)^T \in \mathbb{R}^3$ , is the twist of  $B$ ;  $\mathbf{M} = \text{diag}(m, m, J) \in \mathbb{R}^{3 \times 3}$ , is the inertia matrix with  $m > 0$  and  $J > 0$  being the mass and the rotational inertia of the load, respectively;  $\mathbf{g} \in \mathbb{R}^3$  is the wrench resulting from the environmental forces such as friction or gravitation (in our setting we assume  $\mathbf{g} = \mathbf{0}$ ); and  $\mathbf{u} \in \mathbb{R}^3$  denotes the external wrench applied by the robots to  $B$ , which will be characterized in the following. All the previous quantities are expressed in  $\mathcal{W}$ .

Each robot  $i$  contributes to the manipulation tasks with one end effector. The extension to the more general case of multiple arms per robot is however straightforward. We denote with  $\mathbf{u}_i = (\mathbf{f}_i^T \tau_i)^T \in \mathbb{R}^3$  the wrench exerted by the end-effector of robot  $i$ , expressed in  $\mathcal{W}$ , where  $i = 1 \dots n$ . The force  $\mathbf{f}_i \in \mathbb{R}^2$  is applied to a contact point  $C_i$  of  $B$  and lies on the plane  $\mathbf{x}\mathbf{y}$ , and  $\tau_i \in \mathbb{R}$  is the intensity of a torque applied about the normal direction to the plane  $\mathbf{x}\mathbf{y}$ . We assume, naturally, that contact points do not overlap, i.e.,  $C_i \neq C_j, \forall i, j = 1 \dots n$ .

The total external wrench applied to  $B$  is given by

$$\mathbf{u} = \sum_{i=1}^n \mathbf{G}_i \mathbf{u}_i = \mathbf{G} \bar{\mathbf{u}}, \quad (2)$$

where  $\mathbf{G}_i \in \mathbb{R}^{3 \times 3}$  is the partial grasp matrix,  $\mathbf{G} \in \mathbb{R}^{3 \times 3n}$  is the grasp matrix, and  $\bar{\mathbf{u}} = (\mathbf{u}_1^T \dots \mathbf{u}_n^T)^T$  is the stacked applied wrench that groups the generalized contact force components transmitted through the contact points [12]. The partial grasp matrix is defined as  $\mathbf{G}_i = \mathbf{P}_i \bar{\mathbf{R}}_i$ , where

$$\mathbf{P}_i = \begin{pmatrix} \mathbf{I}_{2 \times 2} & \mathbf{0}_{2 \times 1} \\ [(\mathbf{p}_{C_i} - \mathbf{p}_C)^\perp]^T & 1 \end{pmatrix}, \quad (3)$$

and  $\bar{\mathbf{R}}_i = \mathbf{I}_{3 \times 3}$ , in our setting, for all  $i = 1 \dots n$ . Here  $\mathbf{p}_{C_i} \in \mathbb{R}^2$  is the position of  $C_i$  in  $\mathcal{W}$ . The operator  $(\cdot)^\perp$  is defined by

$$\mathbf{q}^\perp = \underbrace{\mathbf{Q}\mathbf{q}}_{=\mathbf{Q}} = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} q^x \\ q^y \end{pmatrix} = \begin{pmatrix} -q^y \\ q^x \end{pmatrix}, \quad (4)$$

that is to say,  $\mathbf{q}^\perp$  is equal to  $\mathbf{q}$  rotated of an angle of  $\pi/2$ . The dynamics (1) of the manipulated load is then given by

$$\begin{pmatrix} \dot{\mathbf{v}}_C \\ \dot{\omega} \end{pmatrix} = \sum_{i=1}^n \begin{pmatrix} m^{-1} \mathbf{I}_{2 \times 2} & \mathbf{0}_{2 \times 1} \\ J^{-1} [(\mathbf{p}_{C_i} - \mathbf{p}_C)^\perp]^T & J^{-1} \end{pmatrix} \begin{pmatrix} \mathbf{f}_i \\ \tau_i \end{pmatrix}. \quad (5)$$

Let  $\mathbf{p}_G \in \mathbb{R}^2$  represent the position of the geometric center  $G$  of the contact points in  $\mathcal{W}$ , i.e.,

$$\mathbf{p}_G = \sum_{i=1}^n \mathbf{p}_{C_i}.$$

We compactly define  $\mathbf{z}_i = \mathbf{p}_{C_i} - \mathbf{p}_G$ , and  $\mathbf{z}_C = \mathbf{p}_G - \mathbf{p}_C$ . Thus, substituting  $\mathbf{p}_{C_i} - \mathbf{p}_C = \mathbf{z}_i + \mathbf{z}_C$  in (5) we obtain

$$\begin{pmatrix} \dot{\mathbf{v}}_C \\ \dot{\omega} \end{pmatrix} = \sum_{i=1}^n \begin{pmatrix} m^{-1} \mathbf{I}_{2 \times 2} & \mathbf{0}_{2 \times 1} \\ J^{-1} \mathbf{z}_i^\perp{}^T & J^{-1} \end{pmatrix} \begin{pmatrix} \mathbf{f}_i \\ \tau_i \end{pmatrix} + \begin{pmatrix} \mathbf{0}_{2 \times 1} \\ J^{-1} \mathbf{z}_C^\perp{}^T \end{pmatrix} \sum_{i=1}^n \mathbf{f}_i. \quad (6)$$

Based on the dynamics (6) one can be easily convinced that in order to effectively manipulate the load by controlling its velocity  $\mathbf{v}_C$  and angular rate  $\omega$  is of fundamental importance that each robot  $i$  has an estimate of the constant parameters  $m$  and  $J$ , the time-varying vectors  $\mathbf{z}_i(t)$  and  $\mathbf{z}_C(t)$ , and the quantities to be controlled, i.e.,  $\mathbf{v}_C(t)$  and  $\omega(t)$ .

We are now ready to formally state the addressed problem:

**Problem** (Distributed Estimation for Cooperative Manipulation). *Given  $n$  robots communicating through an ad-hoc network and manipulating an unknown load  $B$ ; assume that each robot  $i$  can only*

- 1) *locally measure the velocity  $\mathbf{v}_{C_i}$  of the contact point  $C_i$ ,*
- 2) *locally know the applied wrench  $\mathbf{u}_i$  acting on  $C_i$ ,*
- 3) *communicate with its one-hop neighbors denoted with  $\mathcal{N}_i$ .*

*Design a fully-distributed algorithm such that each robot  $i$  is able to estimate the following six quantities:*

- 1) *the (constant) mass  $m$  of the load,*
- 2) *the (constant) rotational inertia  $J$  of the load,*
- 3) *the (time-varying) relative position  $\mathbf{z}_i(t)$  of the contact point  $C_i$  w.r.t. the contact-points geometric center  $G$ ,*
- 4) *the (time-varying) relative position  $\mathbf{z}_C(t)$  of the CoM of  $B$  with respect to  $G$ ,*
- 5) *the (time-varying) velocity  $\mathbf{v}_C(t)$  of the CoM of  $B$ , and*
- 6) *the (time-varying) angular rate  $\omega(t)$ .*

In this work, we consider a quite strict definition of distributed algorithm that requires that the complexity of the computations performed locally by each robot (in terms of number of elementary operations and size of the input/output data) has to be constant with respect to the number of robots  $n$ , like it is done, e.g., in [13] and other works. A distributed algorithm following this definition is highly scalable with  $n$ .

In the next sections we shall constructively prove that the Problem is solvable as long as the communication network is connected, i.e., it exists a multi-hop communication path from any robot to any other robot.

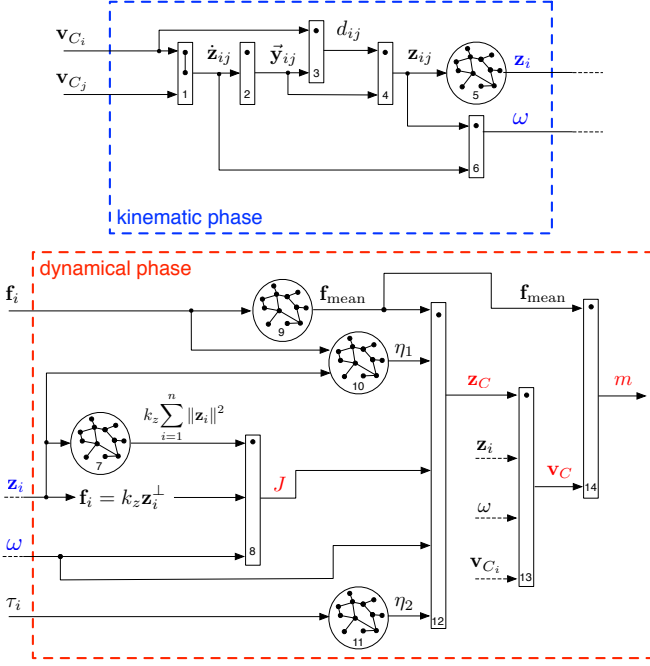


Fig. 2: Overview of the proposed distributed algorithm. **Top** (dashed blue box): this is a *purely kinematic* phase, only the velocity measurements and the rigid body kinematics are used. After this phase, the estimates of the time-varying quantities  $\mathbf{z}_i(t)$  and  $\omega(t)$  (in blue) become available to each robot  $i$ . **Bottom** (dashed red box): this is a *dynamical* phase, also the knowledge of the forces and the rigid body dynamics are used. After this phase, the quantities  $J$ ,  $\mathbf{z}_C(t)$ ,  $\mathbf{v}_C(t)$ , and  $m$  (in red) become available to each robot  $i$ .

### III. ALGORITHM OVERVIEW

An overview of the proposed distributed estimation algorithm is given in the scheme of Fig. 2. Each rectangular box in the scheme corresponds to a computation performed locally by each robot  $i$ . Each circle, instead, corresponds to a consensus-like distributed algorithm that is used to compute the only five global quantities that we shall prove to be needed in the distributed estimation process. The number of these global quantities is independent from the number of robots, and they can be estimated resorting to standard distributed algorithms. Therefore, the overall distributiveness of the approach is ensured. The convergence of these distributed algorithms requires only that the overall communication graph is connected (no all-to-all communication is required). The same applies for our distributed estimation algorithm.

To better understand the overall functioning of the algorithm, it is convenient to think of it as if it is composed by a *purely kinematic* phase, followed by a *dynamical* one. In the former, only the rigid body kinematics constraint and the velocity measurements are used. After this phase, each robot  $i$  is able to estimate the time-varying quantities  $\mathbf{z}_i(t)$  and  $\omega(t)$ . In the latter, the knowledge of the local wrench and the rigid body dynamics is also used. After this phase, each robot  $i$  is able to estimate the remaining quantities, i.e.,  $J$ ,  $\mathbf{z}_C(t)$ ,  $\mathbf{v}_C(t)$ , and  $m$ . The two phases are described in Sections IV and V.

All the estimation blocks are cascaded, therefore convergence or inconsistency issues present in feedback estimation structures are not affecting this scheme.

### IV. KINEMATIC PHASE

The objective of this phase is to distributively compute an estimate of the time-varying quantities  $\mathbf{z}_i(t)$  and  $\omega(t)$ . In the following we shall show how this is possible using only the measured velocities and the rigid body kinematic constraint.

The basic idea of this phase is to split the estimations of  $\mathbf{z}_i(t)$  and  $\omega(t)$  in two parts. The former is common to both the estimations and consists essentially of the estimation of  $\mathbf{z}_{ij}(t) = \mathbf{p}_i(t) - \mathbf{p}_j(t)$ . This part is described in Sec. IV-A. The latter comprises two separate estimators of  $\mathbf{z}_i(t)$  and  $\omega(t)$  and is described in Sec. IV-B.

The reason for passing through the estimation of the  $\mathbf{z}_{ij}$ 's is briefly explained in the following. In [14], a distributed algorithm is proposed that allows the estimation of the centroid of the positions of a network of robots by only measuring the relative positions between pairs of communicating robots. This algorithm can be used to distributively estimate  $\mathbf{z}_i(t)$  if each pair of communicating robots knew the relative position of the contact points  $C_i$  and  $C_j$ , i.e.,  $\mathbf{z}_{ij}(t)$ . Nevertheless, (see Problem II) each robot only measures the velocity of its contact point and not its position. Our first contribution is to show that, thanks to the rigid body constraint, it is possible to estimate  $\mathbf{z}_{ij}(t)$  only resorting to the measures  $\mathbf{v}_i(t)$  and  $\mathbf{v}_j(t)$ .

Hereinafter, we shall avoid to explicitly note the time dependence whenever clear from the context, in order to enhance the text readability.

#### A. Estimation of $\mathbf{z}_{ij}(t)$

The time-varying vector  $\mathbf{z}_{ij}(t)$  that we want to estimate has to obey to the *nonlinear* rigid body constraint

$$\mathbf{z}_{ij}^T \mathbf{z}_{ij} = \text{const.} \quad (7)$$

This leads us to the main idea behind the estimation of  $\mathbf{z}_{ij}(t)$ , that is, to exploit the fact that even if the direction of  $\mathbf{z}_{ij}(t)$  may vary in time, the norm  $\|\mathbf{z}_{ij}\|$  is always constant. Thus, the problem can be broken down into two parts: first, the retrieval of the time-varying direction of  $\mathbf{z}_{ij}(t)$ , and then the estimation of the constant quantity  $\|\mathbf{z}_{ij}\|$ .

The time derivative of both sides of (7) results in

$$\dot{\mathbf{z}}_{ij}^T \mathbf{z}_{ij} = 0, \quad (8)$$

which means that the directions of  $\mathbf{z}_{ij}$  and  $\dot{\mathbf{z}}_{ij}^\perp = \mathbf{Q} \dot{\mathbf{z}}_{ij}$  are the same. We can then explicitly decompose  $\mathbf{z}_{ij}$  in two factors

$$\mathbf{z}_{ij} = d_{ij} \bar{\mathbf{y}}_{ij}, \quad (9)$$

where  $\bar{\mathbf{y}}_{ij} = \dot{\mathbf{z}}_{ij}^\perp / \|\dot{\mathbf{z}}_{ij}\| \in \mathbb{S}^1$  is the unit vector denoting the time-varying oriented line (axis) along which  $\mathbf{z}_{ij}$  lies, and  $d_{ij} \in \mathbb{R}$  is the coordinate of  $\mathbf{z}_{ij}$  on  $\bar{\mathbf{y}}_{ij}$ .

Let each robot send to its neighbors the (measured) velocity of its contact point using the available one-hop communication links. Then, each robot  $i$  can compute the velocity difference

$$\dot{\mathbf{z}}_{ij} = \mathbf{v}_{C_i} - \mathbf{v}_{C_j}, \quad (10)$$

and its orthogonal vector  $\dot{\mathbf{z}}_{ij}^\perp$ , for each  $j \in \mathcal{N}_i$ . As a consequence,  $\bar{\mathbf{y}}_{ij}$  is actually locally available to each robot  $i$ ,  $\forall j \in \mathcal{N}_i$ . This is the first milestone of our algorithm, which is formally stated in the following result.

**Result 1.** The axis  $\vec{y}_{ij}$  along which  $\mathbf{z}_{ij}$  lies is directly computed from local measurements and one-hop communication as

$$\vec{y}_{ij} = \frac{\dot{\mathbf{z}}_{ij}^\perp}{\|\dot{\mathbf{z}}_{ij}^\perp\|} = \frac{(\mathbf{v}_{C_i} - \mathbf{v}_{C_j})^\perp}{\|\mathbf{v}_{C_i} - \mathbf{v}_{C_j}\|}$$

as long as  $\|\dot{\mathbf{z}}_{ij}\| = \|\mathbf{v}_{C_i} - \mathbf{v}_{C_j}\| \neq 0$ .

In order to obtain the sought  $\mathbf{z}_{ij}$ , only the estimation of  $d_{ij}$  is left. Due to the rigid body constraint (7)

$$|d_{ij}| = \|\mathbf{z}_{ij}\| = \text{const} \quad (11)$$

holds, i.e.,  $d_{ij}$  is either equal to  $\|\mathbf{z}_{ij}\|$  or to  $-\|\mathbf{z}_{ij}\|$ , depending on the fact that  $\vec{y}_{ij}$  and  $\mathbf{z}_{ij}$  have the same or the opposite direction. However, since in (9) both  $\mathbf{z}_{ij}(t)$  and  $\vec{y}_{ij}(t)$  are continuous functions of time (for  $\vec{y}_{ij}$  this holds in any open interval in which  $\|\dot{\mathbf{z}}_{ij}\| \neq 0$ ), we have that

$$\text{sign}(d_{ij}) = \text{const} \quad \forall t \in T \quad \text{as long as} \quad \|\dot{\mathbf{z}}_{ij}\| \neq 0 \quad \forall t \in T.$$

Thus, in any time interval  $T$  in which  $\|\dot{\mathbf{z}}_{ij}\| \neq 0$  and under the reasonable assumption that the input wrenches are continuous in  $T$  we can differentiate both sides of (9), thus obtaining

$$\dot{\mathbf{z}}_{ij} = d_{ij} \dot{\vec{y}}_{ij}, \quad (12)$$

which is a linear estimation problem that the robot  $i$  can locally solve to estimate the sought  $d_{ij}$ . In fact, among the quantities that appear in (12), robot  $i$  knows the quantity  $\dot{\mathbf{z}}_{ij}$  and the time integral of  $\frac{d}{dt} \vec{y}_{ij}$ , i.e.,  $\vec{y}_{ij}$ . Therefore, the estimate of  $d_{ij}$  can be carried out resorting to a standard online linear estimation technique described, e.g., in [15], and recapped in the Appendix of [10]. This technique has also the property of averaging out the possible measurement noise. To this aim, the time interval  $T$  can be tuned on the basis of the noise level that has to be averaged out in the velocity measurements.

We observe that after the first estimation of  $d_{ij}$  there is no need to further estimate  $|d_{ij}|$ , since this is a constant quantity. Thus, the only signal to keep track of is  $\text{sign}(d_{ij})$ . This can be easily done instantaneously implementing two linear observers of the dynamic system (12): one assuming  $\text{sign}(d_{ij}) = 1$  and another assuming  $\text{sign}(d_{ij}) = -1$ . Then, it is sufficient to select, at each time-step, the sign of the observer that provides the best estimate in terms of, e.g., measurement residual.

To conclude the description of the algorithm, every time it happens to be  $\|\dot{\mathbf{z}}_{ij}\| = 0$ , the last estimate of  $\mathbf{z}_{ij}$  is kept frozen. In a real implementation the introduction of a suitable threshold to cope with the possible noise is recommended.

We summarize the results of this section in the following.

**Result 2.** The vector  $\mathbf{z}_{ij}$  is estimated locally by robot  $i$  and  $j$  by the separate computation of its two factors

- $\vec{y}_{ij}$  (time-varying axis) computed directly from  $\mathbf{v}_{C_i} - \mathbf{v}_{C_j}$  (see Result 1)
- $d_{ij}$  (constant coordinate) computed from  $\mathbf{v}_{C_i} - \mathbf{v}_{C_j}$  solving (12) via filtering and online linear least squares.

This part of the algorithm is synthetically shown in the blocks 1, 2, 3, and 4 of the diagram of Fig. 2.

## B. Estimation of $\mathbf{z}_i(t)$ and $\omega(t)$

As anticipated before, the estimate of  $\mathbf{z}_{ij}(t)$  provides a straightforward way to estimate  $\mathbf{z}_i$ , as depicted in the block 5 of the diagram of Fig. 2, and recapped in the following

**Result 3.** Once the estimate of  $\mathbf{z}_{ij}(t)$  is available to each robot  $i$ ,  $\forall j \in \mathcal{N}_i$ , each robot  $i$  estimates  $\mathbf{z}_i$  by using the centroid estimation algorithm described in [14].

In order to estimate the angular rate  $\omega$ , we use the following relation from rigid body kinematics

$$\omega \mathbf{z}_{ij} = -\dot{\mathbf{z}}_{ij}^\perp, \quad (13)$$

where  $\dot{\mathbf{z}}_{ij}^\perp$  is computed locally from (10), and  $\mathbf{z}_{ij}$  is locally estimated, as shown in Sec. IV-A. Multiplying both sides of (13) by  $\mathbf{z}_{ij}^T$ , we obtain that, for each pair of communicating robots  $i$  and  $j$ , an estimate of  $\omega$  is directly given by

$$\omega = -\left(\mathbf{z}_{ij}^T \dot{\mathbf{z}}_{ij}^\perp\right) \left(\mathbf{z}_{ij}^T \mathbf{z}_{ij}\right)^{-1}. \quad (14)$$

**Result 4.**  $\omega$  is locally computed using (14), where  $\dot{\mathbf{z}}_{ij}$  comes from direct measurement and one-hop communication and  $\mathbf{z}_{ij}$  from Result 2.

This part of the algorithm is synthetically shown in the block 6 of the diagram of Fig. 2.

The use of (14) provides robot  $i$  with as many estimates of  $\omega$  as the number of its neighbors  $j \in \mathcal{N}_i$ . In case of noiseless velocity measurements all those estimates are identical. In case of noisy velocity measurements, this redundancy can be exploited to average out the noise either at the local level (e.g., by averaging the different estimates corresponding to each neighbor) or at the global level (by, e.g., using some dynamic consensus strategy among all the robots [16]). For the sake of presentation clarity we restrain ourselves from presenting these minor details here.

## V. DYNAMICAL PHASE

The objective of this phase (corresponding to the part boxed by a red dashed line in the block diagram of Fig. 2) is to estimate the remaining quantities, i.e., the (constant) rotational inertia  $J$ , the (time-varying)  $\mathbf{z}_C(t)$  and  $\mathbf{v}_C(t)$ , and the (constant) mass  $m$ . The order in which they are estimated follows a dependency hierarchy. This phase makes use of the velocity measurements  $\mathbf{v}_{C_i}$ , the force inputs  $\mathbf{f}_i$ , as well as the rigid body kinematics and dynamics. The basic operations of this phase are summarized in the following:

- 1) (estimation of  $J$ ) exploit the knowledge of  $\mathbf{z}_i$  to apply a particular input wrench that cancels the effect of  $\mathbf{z}_C$  in (6), thus, obtaining a reduced dynamics in which  $J$  is the only unknown; then, estimate  $J$  using linear least squares;
- 2) (estimation of  $\mathbf{z}_C(t)$ ) use all the previously estimated quantities, the rotational dynamics in (6), and the rigid body constraint to recast the estimation of  $\mathbf{z}_C$  to a nonlinear observation problem that can be solved by each robot resorting to local computation;
- 3) (estimation of  $\mathbf{v}_C(t)$ ) use rigid body kinematics to compute  $\mathbf{v}_C(t)$  from all the quantities estimated so far;
- 4) (estimation of  $m$ ) use a distributed estimation of the total force produced by the robots and  $\mathbf{v}_C(t)$  to finally estimate the constant  $m$  using linear least squares.

### A. Estimation of $J$

Being  $J$  a constant quantity our strategy is to impose a particular wrench for a short time interval in order to let its estimate converge close enough to the real value. After this finite time interval any wrench can be applied again.

Let us isolate the rotational dynamics from (6)

$$\dot{\omega} = \frac{1}{J} \sum_{i=1}^n \mathbf{z}_i^\perp \mathbf{f}_i + \frac{1}{J} \mathbf{z}_C^\perp \sum_{i=1}^n \mathbf{f}_i + \frac{1}{J} \sum_{i=1}^n \tau_i \quad (15)$$

where: (i)  $J$  is the constant to be estimated; (ii)  $\omega(t)$  is locally known by each robot thanks to Result 4; (iii)  $\mathbf{z}_i(t)$  is locally known by each robot thanks to Result 3; (iv)  $\mathbf{f}_i$  and  $\tau_i$  are locally known by each robot, since they are applied by the robot; (v)  $\mathbf{z}_C$  is still unknown. If we were able to eliminate  $\mathbf{z}_C$  from (15), then  $J$  would become the only unknown in (15).

It is easy to verify that the influence of  $\mathbf{z}_C$  in (15) is eliminated if each robot  $i$  applies a force  $\mathbf{f}_i$  such that  $\sum_{i=1}^n \mathbf{f}_i = 0$ . A clever choice is to set  $\mathbf{f}_i = k_z \mathbf{z}_i^\perp$ , where  $k_z \neq 0$  is an arbitrary constant. In fact this choice implies

$$\sum_{i=1}^n \mathbf{f}_i = k_z \sum_{i=1}^n \mathbf{z}_i^\perp = k_z \sum_{i=1}^n (\mathbf{p}_{C_i} - \mathbf{p}_G)^\perp = k_z Q \sum_{i=1}^n (\mathbf{p}_{C_i} - \mathbf{p}_G) = 0.$$

Note that this force can be computed by each robot in a distributed way, since  $\mathbf{z}_i$  is locally known thanks to Result 3.

By applying  $\mathbf{f}_i = k_z \mathbf{z}_i^\perp$  the rotational dynamics (15) becomes

$$\dot{\omega} = \frac{k_z}{J} \sum_{i=1}^n \|\mathbf{z}_i\|^2 + \frac{1}{J} \sum_{i=1}^n \tau_i.$$

In order to further simplify the distributed computation let us impose also  $\tau_i = 0$ ,  $\forall i = 1 \dots n$ , limited to the time interval in which  $J$  is estimated. Hence (15) is further simplified in

$$\dot{\omega} = J^{-1} k_z \sum_{i=1}^n \|\mathbf{z}_i\|^2. \quad (16)$$

Equation (16) expresses a linear relation in which the only unknown is the proportionality factor  $J^{-1}$ . In fact,  $\omega$  and  $k_z \|\mathbf{z}_i\|^2$  are locally known to each robot  $i$ , which means that the constant number  $k_z \sum_{i=1}^n \|\mathbf{z}_i\|^2$  can be computed distributively through an average consensus [19] right after the moment in which each robot is able to estimate  $\mathbf{z}_i$  (block 7 in the diagram of Fig. 2). Therefore, the estimation of  $J$  is recast in (16) as a linear least squares estimation problem that can be solved resorting to the same strategy used to estimate  $d_{ij}$  in (12) (block 8 in the diagram of Fig. 2). A summary follows.

**Result 5.** *Each robot distributively computes the constant sum  $k_z \sum_{i=1}^n \|\mathbf{z}_i\|^2$  using  $\mathbf{z}_i$  from Result 3 and then average consensus. Then, each robot  $i$  applies a force  $\mathbf{f}_i = k_z \mathbf{z}_i^\perp$  for a given time interval, the moment of inertia  $J$  is distributively computed by solving the linear least squares problem (16).*

If noise is present, at this stage each robot may have a slightly different estimate of  $J$ . Thus, a standard average consensus algorithm is executed to average out the noise and improve the estimate of  $J$ .

### B. Estimation of $\mathbf{z}_C$

The main idea behind the estimation of the time-varying quantity  $\mathbf{z}_C(t)$  is to rewrite (15) in order to let only the following kind of quantities appear (in addition to  $\mathbf{z}_C(t)$ ):

- global quantities that can be distributively estimated;
- local quantities available from the problem setting (measures or inputs) or from the previous results.

We shall show the possibility of this rewriting and also that the estimation of  $\mathbf{z}_C(t)$  boils down to a solvable nonlinear observation problem. Let us first decompose the local force  $\mathbf{f}_i(t)$  in two parts and recap two important identities

$$\mathbf{f}_i(t) = \frac{1}{n} \sum_{i=1}^n \mathbf{f}_i(t) + \Delta \mathbf{f}_i(t) = \mathbf{f}_{\text{mean}}(t) + \Delta \mathbf{f}_i(t), \quad (17)$$

$$\sum_{i=1}^n \mathbf{z}_i^\perp = 0 \quad \text{and} \quad \sum_{i=1}^n \Delta \mathbf{f}_i = 0. \quad (18)$$

We can then rewrite (15), exploiting (17) and (18), as

$$\begin{aligned} \dot{\omega} &= \frac{1}{J} \left( \sum_{i=1}^n \mathbf{z}_i^\perp \right) \mathbf{f}_{\text{mean}} + \frac{n}{J} \mathbf{z}_C^\perp \mathbf{f}_{\text{mean}}(t) + \frac{1}{J} \sum_{i=1}^n \mathbf{z}_i^\perp \Delta \mathbf{f}_i + \\ &+ \frac{1}{J} \mathbf{z}_C^\perp \sum_{i=1}^n \Delta \mathbf{f}_i + \frac{1}{J} \sum_{i=1}^n \tau_i = \\ &= \underbrace{\frac{n}{J} \mathbf{z}_C^\perp \mathbf{f}_{\text{mean}}}_{\mathbf{z}_C^\perp \bar{\mathbf{f}}} + \underbrace{\frac{1}{J} \sum_{i=1}^n \mathbf{z}_i^\perp \Delta \mathbf{f}_i}_{\eta_1} + \underbrace{\frac{1}{J} \sum_{i=1}^n \tau_i}_{\eta_2}, \end{aligned}$$

i.e.,

$$\dot{\omega} = \mathbf{z}_C^\perp \bar{\mathbf{f}} + \eta_1 + \eta_2. \quad (19)$$

The global quantities

- $\bar{\mathbf{f}} = \frac{n}{J} \mathbf{f}_{\text{mean}}$ ,
- $\eta_1 = J^{-1} \sum_{i=1}^n \mathbf{z}_i^\perp \Delta \mathbf{f}_i = J^{-1} \sum_{i=1}^n \mathbf{z}_i^\perp \mathbf{f}_i$ , and
- $\eta_2 = J^{-1} \sum_{i=1}^n \tau_i$

can be all distributively estimated in parallel using three instances of the dynamic consensus algorithm [16] (blocks 9, 10, and 11 in the diagram of Fig. 2). The global quantity  $\omega(t)$  is known thanks to Result 4. The only unknown in (19) is  $\mathbf{z}_C(t)$ . Define  $\eta = \eta_1 + \eta_2$ . The following result holds:

**Result 6.** *The rotational dynamics is given by*

$$\dot{\omega} = \mathbf{z}_C^\perp \bar{\mathbf{f}} + \eta \quad (20)$$

where  $\omega(t)$  is known thanks to Result 4 and  $\bar{\mathbf{f}}(t)$  and  $\eta(t)$  are locally known to each robot through distributed computation.

It is clear that, in (20),  $\omega$  and  $\mathbf{z}_C(t)$  play the role of state variables and  $\bar{\mathbf{f}}$  and  $\eta$  are the inputs. In order to complete (20) with the dynamics of  $\mathbf{z}_C(t)$  we recall that  $\mathbf{z}_C$  is a constant-norm vector, rigidly attached to the object, hence

$$\dot{\mathbf{z}}_C = \omega \mathbf{z}_C^\perp. \quad (21)$$

Combining (20) and (21), we obtain the nonlinear system

$$\begin{cases} \dot{x}_1 = -x_2 x_3 \\ \dot{x}_2 = x_1 x_3 \\ \dot{x}_3 = x_1 u_2 - x_2 u_1 + u_3 \\ y = x_3 \end{cases}, \quad (22)$$

where  $\mathbf{z}_C = (\mathbf{z}_C^x \mathbf{z}_C^y)^T = (x_1 \ x_2)^T$  is the unknown part of the state vector,  $\boldsymbol{\omega} = x_3$  is the measured part of the state vector and, consequently, can be considered as the system output, and  $\bar{\mathbf{f}} = (\bar{f}_x \ \bar{f}_y)^T = (u_1 \ u_2)^T$ ,  $\boldsymbol{\eta} = u_3$  are known inputs.

**Result 7.** *Estimating  $\mathbf{z}_C(t)$  is equivalent to observe the state  $(x_1 \ x_2)^T$  of the nonlinear system (22) with known output  $y = x_3 = \boldsymbol{\omega}$  and known inputs  $u_1 = \bar{f}_x$ ,  $u_2 = \bar{f}_y$ , and  $u_3 = \boldsymbol{\eta}$ .*

In [11], we studied the observability of (22):

**Proposition 1.** *If  $x_3 \neq 0$  and  $(u_1 \ u_2)^T \neq \mathbf{0}$ , then system (22) is locally observable in the sense of [17].*

*Proof:* Given in [11]. ■

Note that the applied torques  $\tau_i$ , for  $i = 1 \dots n$  (which are included in  $u_3$ ) have no influence on the observability of  $\mathbf{z}_C(t)$ . In [11], we also proposed a nonlinear observer for the system (22), which is recapped in the following result.

**Proposition 2.** *Consider the following dynamical system*

$$\begin{aligned}\dot{\hat{x}}_1 &= -\hat{x}_2 x_3 + u_2(y - \hat{x}_3) \\ \dot{\hat{x}}_2 &= \hat{x}_1 x_3 - u_1(y - \hat{x}_3) \\ \dot{\hat{x}}_3 &= \hat{x}_1 u_2 - \hat{x}_2 u_1 + k_e(y - \hat{x}_3) + u_3,\end{aligned}\quad (23)$$

where  $k_e > 0$ . If  $y(t) \neq 0$  and  $(u_1(t) \ u_2(t))^T \neq \mathbf{0}$ , then (23) is an asymptotic observer for (22), i.e., defining  $\hat{\mathbf{x}} = (\hat{x}_1 \ \hat{x}_2 \ \hat{x}_3)^T$  and  $\mathbf{x} = (x_1 \ x_2 \ x_3)^T$ , one has that  $\hat{\mathbf{x}}(t) \rightarrow \mathbf{x}(t)$  asymptotically.

*Proof:* Given in [11]. ■

Thanks to Proposition 2 we can state the following result:

**Result 8.** *The relative position of the CoM w.r.t. the center of the grasping points, i.e.,  $\mathbf{z}_C(t)$ , is distributively computed by using the observer (23) and thanks to the local knowledge of  $n$ ,  $J$ ,  $\boldsymbol{\omega}$ ,  $\mathbf{f}_{\text{mean}}$ , and  $\sum_{i=1}^n \mathbf{z}_i^T \Delta \mathbf{f}_i$  from the previous results.*

The just described estimation of  $\mathbf{z}_C(t)$  is briefly shown in the blocks 9,10,11 (dynamic consensus algorithms) and 12 (observer introduced in (23)) of the diagram of Fig. 2.

### C. Estimation of $\mathbf{v}_C$

The velocity of the center of mass  $\mathbf{v}_C(t)$  is estimated locally by each robot  $i$  exploiting the rigid body constraint

$$\frac{d}{dt}(\mathbf{p}_C - \mathbf{p}_{C_i}) = \boldsymbol{\omega}(\mathbf{p}_C - \mathbf{p}_{C_i})^\perp,$$

which can be rewritten as

$$\mathbf{v}_C(t) = \mathbf{v}_{C_i}(t) - \boldsymbol{\omega}(t)(\mathbf{z}_C(t) + \mathbf{z}_i(t))^\perp, \quad (24)$$

whose rhs elements are all known since:

- $\mathbf{v}_{C_i}(t)$  is locally measured by robot  $i$
- $\boldsymbol{\omega}(t)$ ,  $\mathbf{z}_C(t)$ , and  $\mathbf{z}_i(t)$  are known by each robot  $i$  thanks to Results 4, 8, and 3, respectively.

**Result 9.** *The CoM velocity  $\mathbf{v}_C(t)$  is distributively computed using (24) and the knowledge of  $\mathbf{v}_{C_i}(t)$ ,  $\boldsymbol{\omega}(t)$ ,  $\mathbf{z}_C(t)$ ,  $\mathbf{z}_i(t)$ .*

The block 13 in the diagram of Fig. 2 represents this part.

### D. Estimation of $m$

The estimation of the mass  $m$  of the load is a straightforward consequence of the estimation of the velocity and total (average) force. In fact, rewriting (5) as  $\dot{\mathbf{v}}_C = \frac{1}{m} \sum_{i=1}^n \mathbf{f}_i = \frac{n}{m} \mathbf{f}_{\text{mean}}$ , we obtain

$$\dot{\mathbf{v}}_C = m^{-1} n \mathbf{f}_{\text{mean}}, \quad (25)$$

where

- $n$  is known,
- $\mathbf{f}_{\text{mean}}$  is distributively estimated from  $\mathbf{f}_i$  using dynamic consensus (as in Result 6),
- $\mathbf{v}_C$  is known locally by each robot  $i$  thanks to Result 9.

Thus the problem is recast as the linear least square estimation problem (25) that can be solved resorting to the same strategy used to estimate  $d_{ij}$  in (12) and  $J$  in (16).

**Result 10.** *The mass  $m$  is distributively computed from the knowledge of  $\mathbf{v}$  and  $n$ , and  $\mathbf{f}_{\text{mean}}$  by solving an online linear least square problem via filtering (25).*

The block 14 in the diagram of Fig. 2 represents this part.

## VI. CASE STUDY: UTILITY OF SIMPLE LOCAL RULES

In the previous sections, we have shown how to distributively estimate all the quantities that are needed, e.g., to precisely control a planar load with multiple mobile robots. A part from the phase in which  $J$  is estimated (Sec. V-A), in all the other phases we did not suggest any control input to move the load and perform the estimation. The user of the algorithm is free to use any control input, as long as it ensures the observability of the quantities to be estimated. In particular, we have seen that the observability is related to two conditions: non-zero angular rate  $\boldsymbol{\omega}$  and non-zero average force  $\mathbf{f}_{\text{mean}}$ . In each phase, either one or both of the two conditions are needed to ensure a convergent estimation.

In the following, we prove that an extremely basic control strategy satisfies, under very mild conditions, the aforementioned observability requirements. Furthermore this control strategy: (i) can be implemented resorting only to local perception and communication (it is, therefore, distributed); and (ii) does not require the knowledge of that parameters and quantities that are the objectives of the distributed estimation (it is estimation-‘agnostic’). Hence, it can be applied during the estimation process and independently from it.

**Proposition 3.** *Assume that the following local control rule is used:  $\mathbf{f}_i = \mathbf{f}^* = \text{const}$ ,  $\tau_i = 0$ ,  $\forall i = 1 \dots n$ , and denote with  $\omega_0$  the rotational rate at  $t = 0$ , then*

1)  $\boldsymbol{\omega}(t)$  remains bounded, in particular:

$$|\boldsymbol{\omega}(t)| \leq \sqrt{\omega_0^2 + 4nJ^{-1} \|\mathbf{f}^*\| \|\mathbf{z}_C\|} \quad \forall t \geq 0 \quad (26)$$

2)  $\exists \bar{t} \geq 0$  such that  $\boldsymbol{\omega}$  becomes identically 0,  $\forall t \geq \bar{t}$ , if and only if the following condition is verified

$$2n\mathbf{z}_C(0)^T \mathbf{f}^* - J\omega_0^2 = 2n \|\mathbf{z}_C\| \|\mathbf{f}^*\|. \quad (27)$$

Therefore, this control law is suitable for the estimation process a part from the zero measure case provided by (27).

*Proof:* In order to prove (26) consider the following scalar quantity  $\alpha = \omega^2 - 2nJ^{-1} \mathbf{z}_C^T \mathbf{f}^*$ . Let's now take the derivative of

$\alpha$  w.r.t. time. Exploiting (20) and (21) we obtain  $\dot{\alpha} = 0$ , i.e.,  $\alpha$  is an invariant along the system trajectories when  $\mathbf{f}_i = \mathbf{f}^* = \text{const}$ ,  $\forall i = 1 \dots n$ . In particular  $\alpha(t) = \alpha(0)$ , which implies

$$\begin{aligned}\omega^2(t) &= \omega_0^2 - 2nJ^{-1}\mathbf{z}_C^T(0)\mathbf{f}^* + 2nJ^{-1}\mathbf{z}_C^T(t)\mathbf{f}^* \\ &= \omega_0^2 + 2nJ^{-1}(\mathbf{z}_C(t) - \mathbf{z}_C(0))^T\mathbf{f}^* \\ &\leq \omega_0^2 + 2nJ^{-1}\|\mathbf{z}_C(t) - \mathbf{z}_C(0)\|\|\mathbf{f}^*\| \\ &\leq \omega_0^2 + 4nJ^{-1}\|\mathbf{z}_C\|\|\mathbf{f}^*\|,\end{aligned}\quad (28)$$

which proves (26). Note that to derive (29) we used the fact that  $\|\mathbf{z}_C\|$  is constant over time.

In order to prove (27) we impose that  $0 = \omega(\bar{t}) = \dot{\omega}(\bar{t}) = \ddot{\omega}(\bar{t}) = \dots$  for all the derivatives. Imposing that  $\omega(\bar{t}) = 0$  in (28) for  $t = \bar{t}$  we have

$$0 = \omega_0^2 - 2nJ^{-1}\mathbf{z}_C^T(0)\mathbf{f}^* + 2nJ^{-1}\mathbf{z}_C^T(\bar{t})\mathbf{f}^*. \quad (30)$$

Setting  $\dot{\omega}(\bar{t}) = 0$ ,  $\mathbf{f}_i = \mathbf{f}^*$ ,  $\tau_i = 0$ ,  $\forall i = 1 \dots n$  in (19) we obtain

$$\mathbf{z}_C^\perp(\bar{t})^T\mathbf{f}^* = 0, \quad \Rightarrow \quad \mathbf{z}_C^T(\bar{t})\mathbf{f}^* = \|\mathbf{z}_C\|\|\mathbf{f}^*\|. \quad (31)$$

Substituting (31) in (30) gives

$$0 = \omega_0^2 - 2nJ^{-1}\mathbf{z}_C^T(0)\mathbf{f}^* + 2nJ^{-1}\|\mathbf{z}_C\|\|\mathbf{f}^*\|, \quad (32)$$

that, reordered, gives (27). The proof is concluded by noticing that  $\omega(\bar{t}) = \dot{\omega}(\bar{t}) = 0$  implies (see (20) and (21)) that all the higher order derivatives of  $\omega$  at  $\bar{t}$  are zero as well. ■

The use of the simple local control action of Proposition 3 ensures the sought observability conditions under the very mild conditions (27). However, it causes the load CoM velocity to grow linearly over time (see, e.g., (25)). Therefore, it is wise to modify that control action by periodically changing the direction of the common force (i.e., switching between  $\mathbf{f}^*$  and  $-\mathbf{f}^*$  on a periodical basis). In this way, the CoM velocity will oscillate bounded around zero.

It is also important to have available a control strategy that is able to stop the load motion if needed (like, e.g., at the end of all the estimation phases). In the following result we show that another simple control strategy can be effectively used for braking or stopping purposes.

**Proposition 4.** *Assume that the following local control rule is used:  $\mathbf{f}_i = -b\mathbf{v}_{C_i}$ ,  $\tau_i = 0$ ,  $\forall i = 1 \dots n$ , with  $b > 0$ . Then, both  $\omega$  and  $\mathbf{v}_C$  converge asymptotically to zero with a convergence rate that is proportional to  $b$ .*

*Proof:* From (24) and using the identities on the left in (18) it is straightforward to derive the following two identities

$$\sum_{i=1}^n \mathbf{v}_{C_i} = n\mathbf{v}_C + n\omega\mathbf{z}_C^\perp; \quad \mathbf{v}_{C_i} = \mathbf{v}_C + \omega(\mathbf{z}_C + \mathbf{z}_i)^\perp,$$

which can be used to obtain, respectively

$$n\mathbf{f}_{\text{mean}} = -b\sum_{i=1}^n \mathbf{v}_{C_i} = -bn\mathbf{v}_C - nb\omega\mathbf{z}_C^\perp, \quad (33)$$

$$\begin{aligned}\eta &= -\frac{b}{J}\sum_{i=1}^n \mathbf{z}_i^\perp{}^T \mathbf{v}_{C_i} = -\frac{b}{J}\left[ (\mathbf{v}_C + \omega\mathbf{z}_C^\perp)^T \underbrace{\sum_{i=1}^n \mathbf{z}_i^\perp}_{=0} + \omega \sum_{i=1}^n \mathbf{z}_i^\perp{}^T \mathbf{z}_i^\perp \right] \\ &= -\frac{b}{J}\omega \sum_{i=1}^n \|\mathbf{z}_i\|^2.\end{aligned}\quad (34)$$

Plugging (33) and (34) in (20) and (25) we obtain

$$\begin{aligned}J\dot{\omega} &= -bn\left(\mathbf{z}_C^\perp{}^T \mathbf{v}_C + \omega\mathbf{z}_C^\perp{}^T \mathbf{z}_C^\perp\right) - b\omega \sum_{i=1}^n \|\mathbf{z}_i\|^2 \\ m\dot{\mathbf{v}}_C &= -bn\left(\mathbf{v}_C + \omega\mathbf{z}_C^\perp\right).\end{aligned}$$

Take  $V = \frac{J\omega^2 + m\|\mathbf{v}_C\|^2}{2}$  as Lyapunov candidate. We obtain

$$\begin{aligned}\dot{V} &= -bn\left(\mathbf{v}_C^T \mathbf{v}_C + 2\omega\mathbf{z}_C^\perp{}^T \mathbf{v}_C + \omega^2\mathbf{z}_C^\perp{}^T \mathbf{z}_C^\perp\right) - b\omega^2 \sum_{i=1}^n \|\mathbf{z}_i\|^2 = \\ &= -b\left(n\|\mathbf{v}_C + \omega\mathbf{z}_C^\perp\|^2 + \omega^2 \sum_{i=1}^n \|\mathbf{z}_i\|^2\right) < 0 \quad \forall [\mathbf{v}_C^T \omega] \neq \mathbf{0}^T\end{aligned}$$

which proves the thesis of the proposition. ■

## VII. NUMERICAL TEST

In order to give an idea about how the whole algorithm works in practice we made a basic simulation of the estimation algorithm where a planar load with  $m = 50$  kg and  $J = 86.89$  kg m<sup>2</sup> is manipulated by a team of  $n = 10$  mobile manipulators communicating over a simple line-topology network (i.e., the less connected and more challenging one). The velocity measurements are added a zero-mean Gaussian noise with covariance  $\Sigma_i = \sigma^2 \mathbf{I}_{2 \times 2}$ , and  $\sigma = 0.3$  m/s. The signals needed to understand and evaluate the execution of the entire algorithm are shown in Fig. 3.

In the first step each robot applies an arbitrary force and executes the procedure described in Sec. IV-A to estimate the relative distances between contact points. Due to the presence of noise the estimation is kept ‘frozen’ every time the signal-to-noise ratio is too small, i.e., for what concerns this simulation, whenever  $\|\dot{\mathbf{z}}_{ij}\| \leq 0.5$  m/s. The first plot of Fig. 3 reports the convergence to zero of the Estimation Error Relative Distance (EERD) index, which is defined as  $\text{EERD}(t) = \sum_{i=1}^{n-1} \sum_{j=i}^n G(i, j) [(\mathbf{z}_{ij}(t) - \hat{\mathbf{z}}_{ij}(t))^T (\mathbf{z}_{ij}(t) - \hat{\mathbf{z}}_{ij}(t))]^{1/2}$ , where  $\hat{\mathbf{x}}$  is used from now to indicate the estimate of “ $\mathbf{x}$ ”.

Starting from  $t = 10$  s each robot applies the control rules given in the Propositions 3 and 4 that guarantee both the observability and boundedness of  $[\mathbf{v}_C^T \omega]^T$ . At  $t = 10$  s,  $\omega$  and  $\mathbf{z}_i$  start to be estimated, as described in Sec. IV-B. The second and third plots of Fig. 3 report  $\omega$  and  $\hat{\omega}$ , and the Estimation Error CoM relative distance index (EEC), respectively, where  $\text{EEC}(t) = \sum_{i=1}^n [(\mathbf{z}_i(t) - \hat{\mathbf{z}}_i(t))^T (\mathbf{z}_i(t) - \hat{\mathbf{z}}_i(t))]^{1/2}$ .

Subsequently, at  $t = 20$  s, the first step of the dynamical phase is executed, as described in Sec. V-A. First each robot runs an average consensus in order to locally estimate the constant value  $k_z \sum_{i=1}^n \|\mathbf{z}_i\|^2$ . Then, at  $t = 30$  s each robot  $i$  runs a least square estimation of  $J$  using also the knowledge of  $\hat{\omega}$ . Each robot checks the convergence of the least squares estimation evaluating the variance of the estimator [18]. Then, starting at  $t = 40$  s, the local estimates are exchanged over the network and an average consensus is run to agree on a common estimate  $\hat{J} = 85.67$  kg m<sup>2</sup> (fourth plot in Fig. 3). Then the angular rate is brought to zero (Proposition 4).

Afterwards, at  $t = 80$  s each robot starts the nonlinear observation of  $\mathbf{z}_C$  described in Sec. V-B. The observer errors reach zero at about  $t = 135$  s, as visible in the fifth plot of Fig. 3. The estimate  $\hat{\mathbf{v}}_C$  is then computed using (24) (sixth



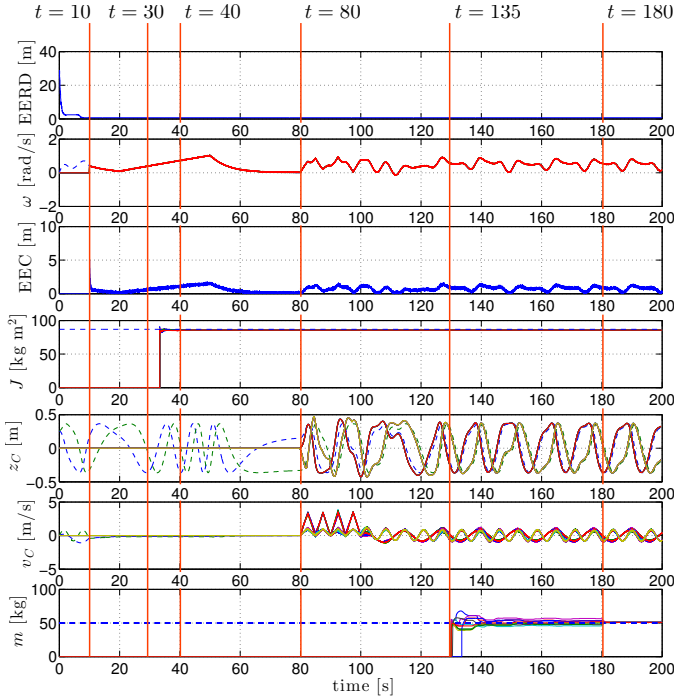


Fig. 3: One illustrative simulation of the whole estimation algorithm described in Secs. IV and V. From top to the bottom, respectively: the trend of the EERD index, the estimates of  $\omega$ , the trend of the EEC index, the estimates of  $J$ , the observations of  $\mathbf{z}_C$ , the estimates of  $\mathbf{v}_C$ , and the estimates of  $m$ .

plot of Fig. 3), which then allows to compute  $\hat{m}$ , as explained in Sec. V-D, by a preliminary collections of samples and local least squares estimations. An average consensus, starting at  $t = 180$ s, allows then to reach an accurate estimate of  $m$  at  $t = 200$ s (seventh and last plot of Fig. 3).

The duration of the entire algorithm is 200s, of which a large portion is needed to collect samples to run the local least squares and the consensus algorithms for the constant parameters  $m$ ,  $J$  and  $d_{ij}$ . The durations of these phases depend on the noise level. Ideally, in absence of noise, a single sample would be sufficient to perform the estimation, while in the real, noisy, case a trade-off between robustness [15] and duration of the estimation phase is requested. Finally, also the convergence time of  $\hat{\mathbf{z}}_C$  can be shortened by acting on the value of  $k_e$  in (23), and the gains of the consensus algorithms can be tuned in order to speed up the agreement [19].

## VIII. CONCLUSIONS

In this paper, we propose a fully-distributed method for the estimation of the parameters needed by a team of ground (planar) mobile robots to collectively manipulate an unknown load. The proposed algorithm provides the estimation of the kinematic and dynamic parameters, as well as the estimate of the kinematic state of the load, i.e., velocity of the center of mass and rotational rate. The approach is totally distributed, and relies on the geometry of the rigid body kinematics, on the rigid body dynamics, on nonlinear observations, and on consensus strategies. It is based on a sequence of steps that is proven to converge in finite time, and at the end of the procedure all the robots will agree on the estimation of

parameters. The only requirements are related to the communication network, that is only required to be connected, and to the capability of each robot to be able to control the local force applied to the load, while measuring the velocity of the contact point. A testing simulation has been run to confirm the effectiveness of our approach.

Extension to the manipulation of 3D objects and experimental tests could be topics to be addressed in future works.

## REFERENCES

- [1] K. I. Kim and Y. F. Zheng, "Two strategies of position and force control for two industrial robots handling a single object," *Robotics and Autonomous Systems*, vol. 5, no. 4, pp. 395–403, 1989.
- [2] S. A. Schneider and R. H. Cannon, "Object impedance control for cooperative manipulation: theory and experimental results," *IEEE Trans. on Robotics and Automation*, vol. 8, no. 3, pp. 383–394, 1992.
- [3] I. D. Walker, R. A. Freeman, and S. I. Marcus, "Analysis of motion and internal loading of objects grasped by multiple cooperating manipulators," *Robotics and Autonomous Systems*, vol. 10, no. 4, pp. 396–409, 1991.
- [4] J. Szewczyk, F. Plumet, and P. Bidaud, "Planning and controlling cooperating robots through distributed impedance," *Journal of Robotic Systems*, vol. 19, no. 6, pp. 283–297, 2002.
- [5] D. Sieber, F. Deroo, and S. Hirche, "Formation-based approach for multi-robot cooperative manipulation based on optimal control design," in *2013 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, Tokyo, Japan, Nov. 2013, pp. 5227–5233.
- [6] Y. Yong, T. Arima, and S. Tsujio, "Inertia parameter estimation of planar object in pushing operation," in *2005 IEEE Int. Conf. on Information Acquisition*, Hong Kong and Macau, China, June 2005, pp. 356–361.
- [7] D. Kubus, T. Kroger, and F. M. Wahl, "On-line estimation of inertial parameters using a recursive total least-squares approach," in *2008 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, Nice, France, Sep. 2008, pp. 3845–3852.
- [8] S. Erhart and S. Hirche, "Adaptive force/velocity control for multi-robot cooperative manipulation under uncertain kinematic parameters," in *2013 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, Tokyo, Japan, Nov. 2013, pp. 307–314.
- [9] J. Markdahl, Y. Karayiannidis, X. Hu, and D. Kragic, "Distributed cooperative object attitude manipulation," in *2012 IEEE Int. Conf. on Robotics and Automation*, St. Paul, MN, May 2012, pp. 2960–2965.
- [10] A. Franchi, A. Petitti, and A. Rizzo, "Distributed estimation of the inertial parameters of an unknown load via multi-robot manipulation," in *53rd IEEE Conf. on Decision and Control*, Los Angeles, CA, Dec. 2014, pp. 6111–6116.
- [11] —, "Decentralized parameter estimation and observation for cooperative mobile manipulation of an unknown load using noisy measurements," in *2015 IEEE Int. Conf. on Robotics and Automation*, Seattle, WA, May 2015, pp. 5517–5522.
- [12] D. Prattichizzo and J. C. Trinkle, "Grasping," in *Springer Handbook of Robotics*, B. Siciliano and O. Khatib, Eds. Springer, 2008, pp. 671–700.
- [13] P. Robuffo Giordano, A. Franchi, C. Secchi, and H. H. Bühlhoff, "A passivity-based decentralized strategy for generalized connectivity maintenance," *The International Journal of Robotics Research*, vol. 32, no. 3, pp. 299–323, 2013.
- [14] R. Aragues, L. Carlone, C. Sagues, and G. Calafiore, "Distributed centroid estimation from noisy relative measurements," *Systems & Control Letters*, vol. 61, no. 7, pp. 773–779, 2012.
- [15] J. J. E. Slotine and W. Li, *Applied nonlinear control*. Prentice Hall, 1991.
- [16] M. Zhu and S. Martinez, "Discrete-time dynamic average consensus," *Automatica*, vol. 46, no. 2, pp. 322–329, 2010.
- [17] R. Hermann and A. J. Krener, "Nonlinear controllability and observability," *IEEE Trans. on Automatic Control*, vol. 22, no. 5, pp. 728–740, 1977.
- [18] S. Van de Geer, "Least squares estimators," ser. Encyclopedia Statistics in The Behavioral Sciences, B. Everitt and D. Howell, Eds. John Wiley, 2005.
- [19] R. Olfati-Saber, J. Fax, and R. Murray, "Consensus and cooperation in networked multi-agent systems," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 215–233, Jan. 2007.